Me&Him

Télémétrie Vision

Version 2.0

Historique des révisions

Date	Version	Description	Auteur
08/10/05	1.0	Première ébauche	Simon Fryer Jean-François Houle
11/10/05	1.1	Section 4 et 5	Jean-François Houle
12/10/05	1.2	Section 5	Jean-François Houle
30/10/05	2.0	Deuxième ébauche basé sur les rencontres 2 et 3 avec le client.	Simon Fryer Jean-François Houle

Table des matières

1. Introduction 5
   - 1.1 Objectif 5
   - 1.2 Portée 5
   - 1.3 Définitions, acronymes et abréviations 5
   - 1.4 Références 5
   - 1.5 Vue d'ensemble 6

2. Positionnement 6
   - 2.1 Opportunité d'affaires 6
   - 2.2 Énoncé du problème 6
   - 2.3 Positionnement du produit 7

3. Descriptions des intervenants et des utilisateurs 7
   - 3.1 Marché Cible 7
   - 3.2 Résumé des intervenants 7
   - 3.3 Résumé des utilisateurs 8
   - 3.4 Environnement utilisateur 8
   - 3.5 Profils des intervenants 8
     - 3.5.1 <Stakeholder Name> **Error! Bookmark not defined.**
   - 3.6 Profils des utilisateurs 10
     - 3.6.1 <User Name> **Error! Bookmark not defined.**
   - 3.7 Principaux besoins des intervenants et utilisateurs 11
   - 3.8 Alternatives et Compétition 12
     - 3.8.1 <aCompetitor> **Error! Bookmark not defined.**
     - 3.8.2 <anotherCompetitor> **Error! Bookmark not de fined.**

4. Vue d'ensemble du produit 12
   - 4.1 Perspective du produit 12
   - 4.2 Principaux avantages 13
   - 4.3 Hypothèses et dépendances 14
   - 4.4 Prévisions de coûts et de prix 14
   - 4.5 Licences et installation 14

5. Caractéristiques du produit 15
   - 5.1 <Une caractéristique> **Error! Bookmark not defined.**
   - 5.2 <Une seconde caractéristique> **Error! Bookmark not defined.**
   - 5.3 … **Error! Bookmark not defined.**

6. Contraintes 17

7. Gammes de qualité 17

8. Attributs des caractéristiques **Error! Bookmark not defined.**

9. Autres exigences du produit 19
   - 9.1 Standards applicables 19
   - 9.2 Exigences du système 19

9.3 Exigences de performance 19
9.4 Exigences environnementales 19

10. Exigences de documentation 19
10.1 Manuel de l'utilisateur 19
10.2 Aide en-ligne 19
10.3 Guides d'installation, de configuration, et fichier à lire 19
10.4 Étiquetage et emballage 19

Annexes 20

A Attributs des caractéristiques **Error! Bookmark not defined.**

Vision

1. Introduction

Ce document de vision présente une vue d'ensemble du logiciel de télémétrie en explorant chacune des perspectives importantes de ce projet. En effet, ce document contient les besoins des utilisateurs, les fonctionnalités ainsi que beaucoup de spécifications supplémentaires. De plus, ce document décrit les différentes parties prenantes qui auront une influence significative sur le projet. Le tout est présenté d'une façon courte, abstraite, lisible et facilement gérable. Le détail, de comment le logiciel de télémétrie va répondre aux besoins énumérés dans ce document, est décrit à l'intérieur des use-case et des spécifications supplémentaires.

1.1 Objectif

Le principal objectif de ce document est d'illustrer de façon courte, précise et lisible les différentes fonctionnalités du logiciel de télémétrie qui viendront répondre aux besoins des parties prenantes de ce projet. En fin de compte, ce document fournit une vision globale du projet en présentant les utilisateurs, les fonctionnalités, les attributs et la documentation associée à la solution logicielle. En conséquence, ce document permet, à toutes les personnes travaillant sur le projet, de partager la même vision du projet.

1.2 Portée

Ce document est une vue d'ensemble des différents paramètres de ce projet : son marché, ses utilisateurs, les besoins auquels il répond ainsi que les fonctionnalités qu'il offre. Afin de conserver ce document facile à comprendre, ce document se doit de rester court, concis et précis. C'est pourquoi les détails spécifiques à chaque fonctionnalité ne sont pas présentés dans ce document mais bien dans un document spécifique à l'analyse des exigences logicielles (SRS).

1.3 Définitions, acronymes et abréviations

Cette section contient les définitions d'acronymes pouvant aider à la compréhension du document de vision.

SRS: Software Requirements Spécifications.

UML: Unified Modeling Language.

ÉTS: École de Technologie Supérieur.

RUP: Rational Unified Process.

S.O.N.I.A : Système d'Opérations Nautique Intelligent et Automne.

Heading : Se diriger vers une position donnée, établir un cap.

Depth : Descendre sous le niveau de l'eau (profondeur).

Strafe : Représente un déplacement latéral.

Pitch : Représente l'angle de plongée avec lequel le sous-marin va descendre en profondeur.

1.4 Références

Rational Unified Process ; École de Technologie Supérieur (ÉTS), Canada

https://cours.ele.etsmtl.ca/documentation/RUP_2003.06.13/index.htm

Site Web du club S.O.N.I.A

http://sonia.etsmtl.ca

Document de présentation (Journal Paper – anglais seulement)

http://sonia.etsmtl.ca/en/ETS_jpaper_2005.pdf

Document de description préliminaire sur le système de télémétrie (fourni avec le laboratoire 2).

1.5 Vue d'ensemble

Voici les sections contenues dans ce document :

1. La position du logiciel de télémétrie.
2. Les utilisateurs ciblés et leurs besoins.
3. Les différents intervenants dans le projet.
4. Une vue du produit par rapport à son environnement.
5. Les principales fonctionnalités du produit.
6. Les contraintes sur le produit.

2. Positionnement

2.1 Opportunité d'affaires

Ce logiciel de télémétrie est développé dans un contexte très particulier qui élimine la possibilité d' opportunité d'affaires. En effet, ce projet est réalisé pour le club étudiant S.O.N.I.A et répond à des besoins très spécialisés pour un sous-marin autonome. De plus, le logiciel de télémétrie doit interfacer avec des composants logiciels très développés maison.

2.2 Énoncé du problème

Le problème de	Les membres du club étudiant S.O.N.I.A aimeraient voir en temps réel les valeurs des paramètres physiques du sous-marin et faire exécuter certaines actions lorsqu'il est dans l'eau sans intervention physique.
affecte	les membres du club étudiant S.O.N.I.A, le serveur AUV4 et le système de communication.
dont l'impact est	que les membres du club étudiant S.O.N.I.A ne sont pas en mesure de tester efficacement le sous-marin. De plus, seulement les membres ayant participés au développement logiciel sont en mesure d'opérer le sous-marin.
Une bonne solution serait	de disposer d'un logiciel permettant de visualiser les valeurs et d'effectuer des opérations sur le sous-marin comportant une

	interface facile d'utilisation.

2.3 Positionnement du produit

Pour	les membres du club étudiant S.O.N.I.A représenté par Félix Pageau et David Mercier.
Qui	désirent posséder une application leur permettant de visualiser les paramètres physiques reliés au mouvement du sous-marin ainsi que la possibilité d'effectuer des opérations de contrôle.
Le logiciel de télémétrie	est un logiciel très spécialisé développé spécifiquement pour les besoins du club étudiant S.O.N.I.A.
Qui	permet d'interfacer avec un serveur de prise de décisions (AUV4). En conséquence, le logiciel de télémétrie peut afficher les données en temps réel et apporter des actions directes sur le sous-marin.
Contrairement à	la situation actuelle où les étudiants du club ne possèdent qu'un serveur de prise de décisions (AUV4) qui est capable de prendre des décisions par lui-même et d'un tableur servant d'interface pour entrer les valeurs cibles et d'activer missions et contrôleurs.
Notre produit	doit s'intégrer facilement aux autres applications du club S.O.N.I.A telles que le serveur AUV4 et le système de comunication. De plus, notre produit doit être le moins dispendieux possible et s'intégrer aux outils de développement actuels de l'équipe S.O.N.I.A

3. Descriptions des intervenants et des utilisateurs

3.1 Marché Cible

Puisque ce logiciel de télémétrie est développé spécifiquement pour répondre aux besoins du club S.O.N.I.A, le marché cible se limite à ce groupe de personnes. En effet, le logiciel de télémétrie est une tâche que le club étudiant S.O.N.I.A désire donner à un contractant externe. La forme de compétition existante dans une pareille situation peut être n'importe quel groupe de développeurs.

3.2 Résumé des intervenants

Nom	Description	Responsabilités
Représentant du club S.O.N.I.A	Cette personne représente le client.	Cette personne représente le client. En conséquence, elle doit participer aux activités nécessitant la présence du client. Aussi, cette personne prend les décisions d'acceptation et est responsable de fournir les renseignements techniques à l'équipe de développement par l'entremise du gestionnaire de projet.

Analyste consultant	Cette personne est une source de référence en ce qui à trait aux activités reliées à l'analyse des besoins et spécifications.	Cette personne a comme responsabilité de vérifier que les processus RUP soient bien suivis et que les plans soient conformes à ce processus. De plus, il agit comme personne ressource en termes d'analyse des besoins et spécifications.
Gestionnaire projet	Cette personne possède une vision globale du projet et voit à son bon déroulement de manière à garantir son succès.	Cette personne a comme responsabilité de voir à ce que le projet soit livré selon l'échéancier. De plus, il est le point de communication central avec le client.

3.3 Résumé des utilisateurs

Nom	Description	Responsabilités	Intervenant
Membres du club étudiant S.O.N.I.A	Les membres du club étudiant sont les seuls utilisateurs du logiciel de télémétrie. Ces personnes se serviront de l'application lors de tests, pour des présentations ou bien en compétion. Ces personnes possèdent des expertises très différentes les unes des autres.	Les utilisateurs ont comme responsabilités de participer aux tests de l'application ainsi que de fournir des renseignements techniques à la demande du capitaine du club.	Les membres du club sont représentés par le directeur technique (David Mercier) et par le capitaine (Félix Pageau).

3.4 Environnement utilisateur

Cette section du document de vision présente les caractéristiques de l'environnement de production du logiciel de télémétrie :

- Le logiciel de télémétrie serra utilisé par les membres du club étudiant S.O.N.I.A étant tous de jeunes adultes étudiant à l'ÉTS dans des disciplines de génie différentes.
- Le logiciel de télémétrie sera utilisé sur des ordinateurs portables lors de tests en bassins donc possiblement à l'extérieur.
- Les membres du club possèdent des machines opérant sous windows et linux.
- Le logiciel de télémétrie doit être en mesure d'interfacer parfaitement avec les outils déjà en place dans l'équipe.
- Le logiciel de télémétrie serra utilisé lors de démonstrations à des commanditaires potentiels. En conséquence, il doit être facile à opérer et comprendre. De plus, il doit être attrayant (tape à l'oeil) car les membres de S.O.N.I.A voudront impressionner ces personnes.
- Le club S.O.N.I.A développe déjà du logiciel en Java et veulent un script de compilation Ant afin d'intégrer le logiciel de télémétrie à leurs outils de compilation automatique.

3.5 Profils des intervenants

3.5.1 Représentants du club S.O.N.I.A

Représentant	Du club étudiant S.O.N.I.A. Ce rôle est assuré par Félix Pageau et David Mercier.
Description	Ces personnes sont des représentants du client.
Type	Ces personnes sont membres actifs de S.O.N.I.A et possèdent une vue d'ensemble du sous-marin. En effet, ces personnes participent régulièrement à des présentations du sous-marin. En conséquence, ils doivent être en mesure de répondre à tous les types de questions pouvant survenir. Ils sont tous les deux étudiants en génie logiciel à l'ÉTS.
Responsabilités	Ces personnes représentent le client. En conséquence, elles doivent participer aux activités nécessitant normalement la présence du client. Aussi, ces personnes prennent les décisions d'acceptation et sont responsables de fournir les renseignements techniques à l'équipe de développement par l'entremise du gestionnaire de projet.
Critères de succès	Les représentants du client reconnaissent le succès du projet si tous les besoins des clients sont satisfaits. Par exemple, le logiciel doit être le moins dispendieux possible et doit posséder un script de compilation Ant.
Implication	Les représentants du client communiquent surtout avec le gestionnaire de projet et les analystes. De plus, ils doivent participer aux revues et activités planifiées.
Biens livrables	Les représentants du client doivent fournir des documents techniques sur le sous-marin en général à la demande de l'équipe de développement.
Commentaires / Préoccupations	Les représentants sont aussi des membres actifs du club étudiant à titre de développeurs et sont étudiants en génie logiciel.

3.5.2 Analyste consultant

Représentant	L'analyste consultant est Pierre Bourque.
Description	Cette personne possède une vision globale du projet et voit à son bon déroulement de manière à garantir son succès.
Type	Cette personne possède un background en informatique et de l'expérience dans le domaine de la gestion de projets en informatique.
Responsabilités	L'analyste consultant a comme responsabilité de voir à ce que le logiciel de télémétrie soit développé selon le RUP. Pour ce faire, il participera à des revues des différents artéfacts associés au RUP. De plus, il répondra aux questions de l'équipe de développement.
Critères de succès	L'analyste considère le projet comme un succès si le processus suivi est conforme à ce qu'indique le RUP.
Implication	L'analyste consultant est impliqué dans le projet par l'entremise de revues et de questions.
Biens livrables	Les seules biens livrables requis de l'analyste consultant sont ceux associés aux revues.
Commentaires / Préoccupations	Pierre Bourque est enseignant du cours LOG-410 dans lequel est réalisé l'analyse du logiciel de télémétrie.

3.5.3 Gestionnaire de projet

Représentant	Le rôle du gestionnaire de projet est assuré par M. X.
Description	Ces personnes sont des représentants du client.
Type	Ces personnes sont membres actifs de S.O.N.I.A et possèdent une vue d'ensemble du sous-marin. En effet, ces personnes participent régulièrement à des présentations du sous-marin. En conséquence, elles doivent être en mesure de répondre à tous les types de questions pouvant survenir. Ils sont tous les deux étudiants en génie logiciel à l'ÉTS.
Responsabilités	Cette personne a comme responsabilité de voir à ce que le projet soit livré selon l'échéancier. De plus, il est le point de communication central avec le client.
Critères de succès	Le projet est un succès pour le gestionnaire de projet s'il est livré à temps et si les besoins du client sont comblés.
Implication	Le gestionnaire réalise le plan de projet logiciel, participe à la rédaction du vision ainsi qu'aux autres artéfacts demandés par le RUP. De plus, cette personne participe au revues planifiées et sert d'intermédiare entre le client et l'équipe de développpement.
Biens livrables	Le gestionnaire de projet doit réaliser le plan de projet logiciel.
Commentaires / Préoccupations	

3.6 Profils des utilisateurs

Comme l'application de télémétrie sert à visualiser des informations provenant du serveur AUV4 en temps réel ainsi qu'à porter des actions concrètes sur le sous-marin, tous les membres de S.O.N.I.A sont appelés à se servir de l'application.

3.6.1 Membres du club étudiant S.O.N.I.A

Représentant	Les membres qui participent à la construction du sous-marin.
Description	Les membres du club S.O.N .I.A vont utiliser l'application lors de tests en bassin. Ainsi, ils seront en mesure de visualiser l'état de plusieurs valeurs clés provenant du serveur AUV4 et d'envoyer des commandes au sous-marin.
Type	Ces personnes sont de jeunes adultes étudiant à l'ÉTS dans différentes disciplines de génie. Principalement, ces personnes étudient en génie électrique, mécanique ou logiciel.
Responsabilités	Les membres de S.O.N.I.A seront amenés à tester le logiciel de télémétrie et à fournir des commentaires constructifs à l'équipe de développement. Les membres du club seront aussi responsables de rapporter les fautes trouvées dans l'application à l'équipe de développement suivant les directives indiquées par le contractant.
Critères de succès	Les membres de S.O.N.I.A considèrent le projet réussi si l'application répond à leurs besoins et permet d'améliorer leurs séances de tests. Présentement, les membres du club qui ne participent pas au développement logiciel ne sont pas

	capables d'opérer le sous-marin.
Implication	Des représentants, des utilisateurs doivent participer aux revues, réaliser des évaluations de versions intermédiaires et participer à l'évaluation des fonctions d'aide du système. De plus, certains membres du club peuvent être appelés à répondre à des questions techniques concernant des parties complexes du sous-marin.
Biens livrables	Les membres du club doivent fournir leurs commentaires concernant l'application, par écrit, au gestionnaire de projet.
Commentaires / Préoccupations	Être capable d'opérer le sous-marin sans le support d'un étudiant en génie logiciel.

3.7 Principaux besoins des intervenants et utilisateurs

Besoin	Priorité	Préoccupations	Solution actuelle	Solution Proposée
Effectuer des actions concrètes sur le sous-marin telles qu'activer/désactiver chacun des contrôleurs, assigner une valeur cible aux contrôleurs, etc.	1	Présentement, il n'existe aucune solution permettant d'interragir dynamiquement avec le sous-marin. Donc, l'équipe perd du temps à créer des missions afin de tester des actions spécifiques.	Le club étudiant S.O.N.I.A possède un tableur dans lequel ils peuvent cocher les contrôleurs actifs avant le début du test.	Créer un logiciel permettant de porter des actions dynamiques sur le sous-marin.
Activer dynamiquement des modules de missions.	2	Il n'existe aucune solution permettant d'activer une mission dynamiquement ce qui oblige les membres à attendre la fin d'une mission pour en commencer une autre. Le temps de test pourrait être nettement amélioré.	Le club étudiant S.O.N.I.A possède un tableur dans lequel ils peuvent cocher les modules de mission avant le début du test.	Créer un logiciel permettant d'activer dynamiquement un module de mission.
Afficher en temps réel les informations recues par le serveur AUV4.	1	Pouvoir consulter les données physiques provenant de AUV4 afin d'améliorer les scéances de tests. Ainsi, les membres pourront sauver du temps en comprenant les	Le club S.O.N.I.A ne possède aucun moyen de visualiser dynamiquement les informations d'AUV4.	Créer un logiciel permettant de visualiser cette information.

		réactions du sous-marin.		

3.8 Alternatives et Compétition

Le logiciel de télémétrie étant développé par une équipe d'étudiants bénévoles, il n'y a pas lieu de vérifier les alternatives et compétiteurs potentiels.

4. Vue d'ensemble du produit

Cette section est dédiée à décrire les fonctionnalités du système dans son emsemble. C'est-à-dire ses capacités, ainsi que des moyens de communication entre le système et le logiciel de Télémétrie, de même que sa configuration. Nous émettrons également certaines hypothèses qui nous permettront d'avancer dans le développement du logiciel de Télémétrie.

.

4.1 Perspective du produit

Le logiciel de Télémétrie est un produit qui sera relié au sous-marin S.O.N.I.A par le biais d'une communication sans-fil Wi-Fi. Il est donc intégré au système de pilotage du sous-marin, mais il sera externe physiquement à celui-ci. Ceci implique donc que le sous-marin lui-même devra offrir une connection sans-fil au logiciel de Télémétrie. De ce fait, le logiciel de Télémétrie sera en relation directe avec la partie logicielle déjà existante du sous-marin. Pour réaliser ce projet, nous supposons donc que toute la partie télécommunication du système entier est déjà présente et fonctionnelle.

Comme le logiciel de Télémétrie est susceptible de fonctionner sur les plateformes Linux et Windows, nous avons opté pour un développement en langage de programmation Java. En effet, le système logiciel déjà intégré au sous-marin fonctionne actuellement sous Linux.

La figure 4.1.1 illustre l'architecture globale du système du sous-marin S.O.N.I.A. Nous pouvons facilement y observer la communication sans-fil telle que décrite plus haut.

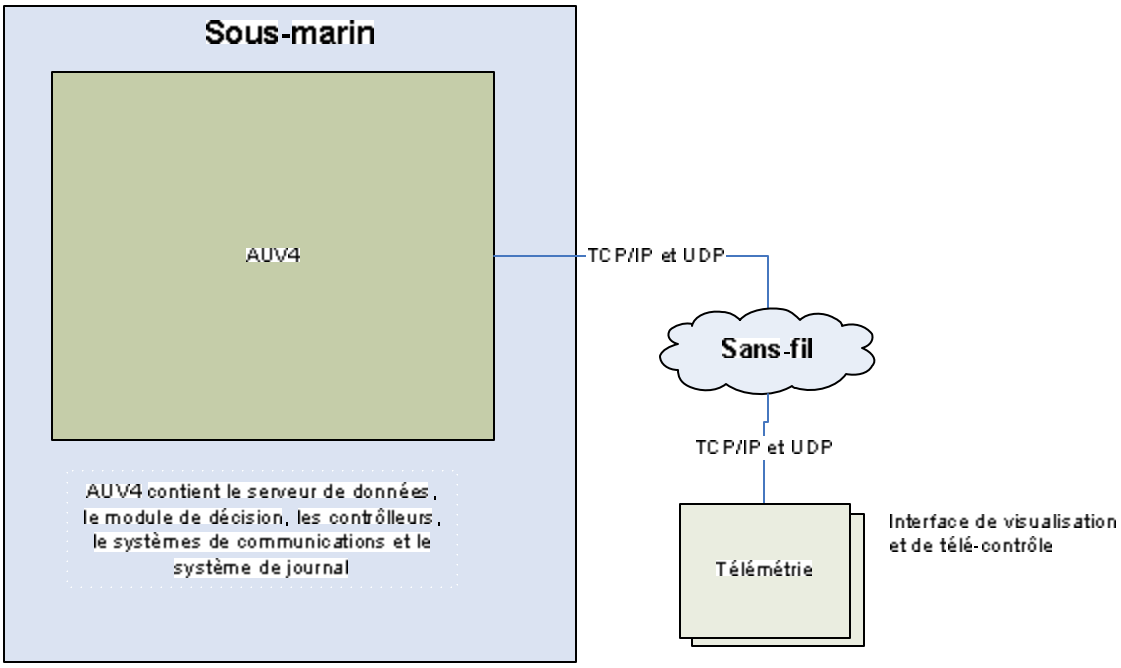
Sous-marin
AUV4
TCP/IP et UDP
Sans-fil
TCP/IP et UDP
Interface de visualisation et de télé-contrôle
Télémétrie
AUV4 contient le serveur de données, le module de décision, les contrôlleurs, le systèmes de communications et le système de journal

Figure 4.1.1: Architecture du système

4.2 Principaux avantages

Un des principaux objectifs du logiciel de Télémétrie est de permettre aux opérateurs du sous-marin, c'est-à-dire les membres de l'équipe S.O.N.I.A, de diriger le sous-marin à distance pour pouvoir effectuer certaines manœuvres de façon ponctuelle. Il permettra donc d'analyser la situation actuelle du sous-marin et de réagir en fonction de celle-ci. Pour parvenir à cet objectif, le logiciel de télémétrie devra supporter trois grandes fonctionnalités.

Voici ces trois fonctionnalités :

- Interpréter les données provenant du sous-marin
- Contrôler les moteurs du sous-marin
- Permettre de créer des missions au sous-marin

Table 4-1 Système de télémétrie

Bénéfices pour le client	Caractéristiques correspondantes
Effectuer des actions concrètes sur le sous-marin telles qu'activer/désactiver chacun des contrôleurs, assigner une valeur cible aux contrôleurs, etc.	• carac-5 • carac-6 • carac-7 • carac-8 • carac-9 • carac-10 • carac-11 • carac-12

Activer dynamiquement des modules de missions.	• carac-12 • carac-13 • carac-15
Afficher en temps réel les informations reçues par le serveur AUV4. Accès direct aux données du sous-marin.	• carac-1 • carac-2 • carac-3 • carac-4 • carac-12 • carac-13

4.3 Hypothèses et dépendances

Pour réaliser le logiciel de Télémétrie, nous devons assumer et poser certaines hypothèses. Ces hypothèses nous permettront de poursuivre le développement du logiciel de télémétrie et ainsi s'assurer d'une complète compréhension des éléments quelque peu ambigus. Voici donc ces hypothèses :

- Nous présumons que l'unité « lali » est un terme connu dans le milieu du sous-marin S.O.N.I.A et que tous les membres de l'équipe savent comment réagir avec cette unité de mesure qui n'est pas « SI ».
- Nous présumons que les matériaux requis pour le système de télécommunication sans fil entre le sous-marin (serveur AUV4) et le logiciel de Télémétrie sont disponibles à moindres coûts selon le budget de l'équipe.
- Nous présumons que nous disposons d'un ordinateur portable assez puissant pour interpréter toute l'information reçue et envoyée par le sous-marin S.O.N.I.A.
- Nous présumons que l'équipe du sous-marin pourra fournir les ressources nécessaires au développement logiciel.
- Le système doit être mis en place et être fonctionnel pour la compétition de l'AUVSI (Association of Unmanned Vehicle Systems International), soit en juillet 2006.

4.4 Prévisions de coûts et de prix

Dû au fait que le logiciel est produit dans un cadre académique et que le budget réel de l'équipe S.O.N.I.A n'est pas divulgué, tous les coûts engendrés par la conception du logiciel ne seront pas calculés. En réalité, un des besoins les plus importants du client est que le logiciel de télémétrie coûte le moins cher possible.

4.5 Licences et installation

Aucune licence n'est applicable pour le logiciel de Télémétrie.

L'installation du logiciel de Télémétrie se fera manuellement sur un seul poste avec une plateforme Windows (compatible Linux également). Il faudra également s'assurer que la machine virtuelle java JRE 1.4.2 ou plus récent est installée sur le poste en question.

Le club S.O.N.I.A se verra également livré le code source de l'application comportant un script de compilation Ant de manière à ce que les membres du club puissent maintenir le logiciel et l'intégrer à leurs outils de compilation automatique.

5. Caractéristiques du produit

Dans cette section nous décrivons toutes les caractéristiques du produit dans son ensemble. Ces caractéristiques représentent la base même du produit. Elles décrivent en détail tout ce que le logiciel de Télémétrie doit pouvoir faire comme opérations et offrir comme fonctionnalités.

5.1 Interprétation de la boussole (carac-1)

Le logiciel de Télémétrie doit pouvoir afficher les données La boussole (anglais : heading) en temps réel. La boussole indique la direction du sous-marin par rapport au nord magnétique.

5.2 Interprétation des propulseurs (carac-2)

Le logiciel de Télémétrie doit pouvoir afficher les données des six propulseurs (anglais : thrusters) du sous-marin en temps réel. Ces données peuvent varier entre -100% et 100%.

5.3 Interprétation du profondimètre (carac-3)

Le logiciel de Télémétrie doit être en mesure d'afficher les valeurs du profondimètre (anglais : depthmeter) en temps réel. Les données envoyées par AUV4 concernant le profondimètre sont affichées en unités de mesure « Lali ». Veuillez vous référer au dictionnaire de terminologie pour une description détaillée du « Lali ».

5.4 Interprétation de la balise acoustique (carac-4)

Le logiciel de Télémétrie doit pouvoir afficher en temps réel les valeurs de la balise acoustique (anglais : hydrophone). Les valeurs résultantes de cette balise, envoyées par AUV4, doivent être affichées en degrés.

5.5 Accès à un contrôleur de cap (carac-5)

Les contrôleurs sont des pilotes automatiques pour les moteurs. On leur assigne une valeur cible et ils doivent diriger le sous-marin pour y parvenir. Dans le cas du contrôleur de cap, le logiciel de Télémétrie doit pouvoir prendre en charge un cap (anglais : heading) désiré et s'y rendre. Pour que le contrôleur fonctionne, le gestionnaire de contrôleurs doit être activé.

5.6 Accès à un contrôleur de profondeur (carac-6)

Le logiciel de Télémétrie doit pouvoir contrôler la profondeur (anglais : depth) du sous-marin. Pour ce faire, l'utilisateur du logiciel de Télémétrie spécifie une valeur cible pour la profondeur et le contrôleur prend en charge les moteurs pour s'y rendre. Pour que le contrôleur fonctionne, le gestionnaire de contrôleurs doit être activé.

5.7 Accès à un contrôleur de déplacements latéraux (carac-7)

Le logiciel de Télémétrie doit pouvoir contrôler les déplacements latéraux (anglais : strafe) du sous-marin. L'utilisateur du logiciel de Télémétrie spécifie une valeur cible pour les déplacements latéraux et le contrôleur prend en charge les moteurs pour s'y rendre. Pour que le contrôleur fonctionne, le gestionnaire de contrôleurs doit être activé.

5.8 Accès à un contrôleur de descente (carac-8)

Le logiciel de Télémétrie doit pouvoir contrôler la descente (anglais : pitch) du sous-marin. Pour ce faire, l'utilisateur du logiciel de Télémétrie spécifie une valeur cible pour la descente et le contrôleur prend en charge les moteurs pour descendre à celle-ci. Pour que le contrôleur fonctionne, le gestionnaire de contrôleurs doit être activé.

5.9 Activer / désactiver chaque contrôleur individuellement (carac-9)

Le logiciel de Télémétrie doit pouvoir activer ou désactiver chaque contrôleur. Si le contrôleur est désactivé, alors le contrôleur en question n'effectuera plus aucune action, même si une valeur cible y a été assignée. Voici la liste des quatre contrôleurs que l'on peut activer ou désactiver :

- Contrôleur de cap (heading)
- Contrôleur de profondeur (depth)
- Contrôleur de déplacements latéraux (strafe)
- Contrôleur de descente (pitch)

5.10 Accès au gestionnaire de contrôleurs (Controller Manager) (carac-10)

Le logiciel de Télémétrie devra avoir un gestionnaire de contrôleurs. Un gestionnaire de contrôleurs est une protection pour autoriser/ empêcher l'utilisation des contrôleurs moteurs. Quand le gestionnaire est activé, aucun contrôleur ne peut fonctionner.

5.11 Contrôle individuel des propulseurs (carac-11)

Le logiciel de Télémétrie doit être en mesure de contrôler individuellement chaque propulseur (anglais : thruster). De ce fait, l'utilisateur spécifie une valeur précise à un moteur en particulier pour que le sous-marin se déplace en contrôle manuel. Pour que le contrôle individuel des propulseurs fonctionne, le gestionnaire de contrôleurs doit être désactivé.

5.12 Interfaces graphiques conviviales (carac-12)

Le logiciel de Télémétrie doit permettre d'effectuer les actions importantes rapidement sans problème. C'est-à-dire qu'en cas de crise, il doit être facile et rapide d'arrêter le sous-marin ou d'annuler les actions prises. De plus, le logiciel de télémétrie doit être fonctionnel sur un ordinateur portable et sans souris externe. C'est-à -dire, avec un « touch pad » ou bien un « track stick ». Par ailleurs, l'interface de télémétrie doit être « tape à l'oeil » car le club S.O.N.I.A va utiliser cette interface afin de contrôler le sous-marin lors de présentations servant à obtenir de nouveaux commanditaires.

5.13 Portable (carac-13)

Le logiciel de Télémétrie doit fonctionner dans plusieurs environnements. Comme le sous-marin S.O.N.I.A est en constante évolution, il se doit de s'adapter aux nouvelles technologies et aux nouveaux environnements. C'est pour ces raisons que le logiciel de Télémétrie devra être développé en Java.

5.14 Doit s'intégrer facilement aux autres composantes du club S.O.N.I.A (carac-14)

Le logiciel de Télémétrie doit pouvoir s'intégrer aux autres modules logiciels ou matériels du sous-marin. Voici la liste des applications du club S.O.N.I.A les plus susceptibles d'être affectées par le logiciel de Télémétrie :

- Simulateur de sous-marin. Composant matériel qui se branche dans un port RS-232 d'un ordinateur (probablement le même que celui utilisé par le logiciel de Télémétrie) et qui simule le mouvement et réponse du sous-marin S.O.N.I.A.
- Logiciel de simulation. Composante logicielle qui simule le sous-marin S.O.N.I.A dans une application graphique. Le logiciel de Télémétrie aura probablement à se brancher sur ce simulateur pour tester ses fonctionnalités.
- Serveur AUV4. C'est ce composant logiciel qui envoie les données provenant des *devices* du sous-marin. Il aura également à recevoir et exécuter les actions envoyées par le logiciel de Télémétrie.

5.15 Accès facile aux modules de missions (carac-15)

Le logiciel de Télémétrie doit pouvoir facilement activer des missions pour le sous-marin. De plus, seulement une

mission à la fois doit pouvoir être active. Ce contrôle se fera au niveau du logiciel de Télémétrie. Une mission est une suite d'actions que doit effectuer le sous-marin afin d'arriver à un certain résultat. Les missions sont à la base même d'une compétition. Le logiciel doit être en mesure de démarrer des missions et de les arrêter si l'utilisateur le désire. Il est important de mentionner qu'un module de mission a priorité sur n'importe quelle autre commande.

6. Contraintes

La contrainte la plus importante est que le logiciel de Télémétrie doit être parfaitement compatible avec AUV4 ansi que tous les outils utilisés par le club S.O.N.I.A. En effet, le logiciel de Télémétrie doit afficher des données provenant de AUV4 et envoyer des commandes à celui-ci. De plus, le logiciel de Télémétrie doit utiliser le système de communication sans-fil ansi que les protocoles fournis par le club S.O.N.I.A. De même, le logiciel doit pouvoir être utilisé avec un portable et sans souris, car les essais se font souvent dans des endroits où il n'y a pas de bureau. En outre, comme ce logiciel est destiné à un club étudiant, il doit coûter le moins cher possible.

7. Gammes de qualité

Un critère de qualité important est que l'interface du logiciel doit être usabile. C'est-à-dire, que l'interface doit être convivial et que les données doivent être présentées de manière à ce qu'elles puissent être facilement et rapidement consultées. En conséquence, suite à l'analyse des informations, des actions pourront être rapidement appliquées au sous-marin. De plus, le code source doit être facilement maintenable. Pour ce faire, les normes de programmation de Sun Microsystems inc. devront être appliquées.

8. Attributs des caractéristiques

Veuillez noter que l'attribut itération fait référence à l'itération dans laquelle sera construite la caractéristique. Pour plus de renseignement sur les itérations, veuillez vous référer au plan de projet SDP-SONIA-TELE-100.

Caractéristiques	***État***	***Bénéfice***	***Effort***	***Risque***	***Stabilité***	***Priorité***	***Itération***
Interprétation de la boussole (carac-1)	Approuvé	Haut	Moyen	Moyen	Stable	Critique	4
Interprétation des propulseurs (carac-2)	Approuvé	Haut	Moyen	Moyen	Stable	Critique	4
Interprétation du profondimètre (carac-3)	Approuvé	Haut	Moyen	Moyen	Stable	Critique	4
Interprétation de la balise acoustique (carac-4)	Approuvé	Haut	Moyen	Moyen	Stable	Critique	4
Accès à un contrôleur de cap (carac-5)	Approuvé	Haut	Moyen	Moyen	Stable	Important	5
Accès à un contrôleur de profondeur (carac-6)	Approuvé	Haut	Moyen	Moyen	Stable	Important	5
Accès à un contrôleur de déplacements latéraux (carac-7)	Approuvé	Haut	Moyen	Moyen	Stable	Important	5
Accès à un contrôleur de descente (carac-8)	Approuvé	Haut	Moyen	Moyen	Stable	Important	5
Activer / désactiver chaque contrôleur individuellement (carac-9)	Approuvé	Haut	Haut	Bas	Stable	Important	5
Accès au gestionnaire de contrôleurs (Controller Manager) (carac-10)	Approuvé	Haut	Haut	Bas	Stable	Important	5
Contrôle individuel des propulseurs (carac-11)	Approuvé	Haut	Haut	Moyen	Stable	Important	4
Interfaces graphiques conviviales (carac-12)	Proposé	Haut	Bas	Bas	Stable	Important	4-5
Portable (carac-13)	Proposé	Moyen	Bas	Bas	Stable	Important	2-4-5
Doit s'intégrer facilement aux autres composantes du club S.O.N.I.A (carac14)	Approuvé	Moyen	Bas	Élevé	Moyen	Critique	2-4
Accès facile aux modules de missions (carac-15)	Approuvé	Moyen	Bas	Moyen	Stable	Important	5

9. Autres exigences du produit

9.1 Standards applicables

Le seul standard applicable est celui de Sun Microsystems inc. concernant la norme de programmation de manière à ce que le logiciel soit plus facile à maintenir par les membres du club S.O.N.I.A.

9.2 Exigences du système

L'utilisateur du logiciel de Télémétrie devra avoir à sa disposition un portable avec un système d'exploitation ayant le Java Runtime. De plus, l'ordinateur devra posséder une carte réseau sans fil.

9.3 Exigences de performance

Le logiciel de Télémétrie doit afficher instantanément les données reçues via le réseau sans fil. Autrement dit, il ne doit pas avoir de « lag » entre la réception des données et l'affichage de celles-ci. De plus, les fonctionnalités livrées au client doivent être totalement fonctionnelles. Il a été mentionné par le client qu'il préférait avoir moins de fonctionnalités que d'en avoir plus qui fonctionnent partiellement.

9.4 Exigences environnementales

Le logiciel de Télémétrie devra fonctionner sur un ordinateur portable de même que sans souris ni clavier autre que celui du portable. De plus, le logiciel doit être facilement maintenable par les membres du club étudiant, car le sous-marin évolue constamment.

10. Exigences de documentation

10.1 Manuel de l'utilisateur

Aucun manuel d'utilisateur ne sera offert.

Nous croyons que seule une interface usabile et des « ToolTipText » seront suffisants.

10.2 Aide en ligne

Aucune aide en ligne ne sera disponible.

10.3 Guides d'installation, de configuration, et fichier à lire

Un guide d'installation sera disponible. Ce guide est l'unique forme de documentation que recevra le client. Il contiendra la procédure de déploiement ainsi qu'une liste des fautes connues.

10.4 Étiquetage et emballage

Il n'y a aucune activité d'emballage et d'étiquetage prévue.

Annexes

A Attributs des caractéristiques

Ces attributs sont assignés à chacune des caractéristiques identifiées précédemment (voir section 8).

État

Proposé	Fonctionnalité qui est uniquement proposée par une ou l'autre des parties. Une fonctionnalité proposée devra faire état d'un suivi dans le futur.
Approuvé	Fonctionnalité qui est acceptée par le client et le chef de projet. Les deux parties sont d'accord pour implanter cette fonctionnalité.
Incorporé	Fonctionnalité qui a déjà été implantée lors d'une itération précédente.

Bénéfice

Haut	Fonctionnalité essentielle. L'échec de l'implémentation signifie que le système ne répondra pas aux besoins du client.
Moyen	Fonctionnalité importante. L'échec de l'implémentation d'une telle fonctionnalité demandra au client de décider s'il veut implémenter cette fonctionnalité ou non. Si le client décide de l'implémenter, il devra replanifier la livraison de cette fonctionnalité avec le chef de projet.
Bas	Fonctionnalité utile. L'échec de l'implémentation n'est pas significatif. Le système pourra être livré comme prévu. Toutefois, le client devra recevoir une liste exaustive des fonctionnalités non-incluses.

Effort

Haut	Lorsque cela implique le travail de 5 à 6 collaborateurs. (Analystes, développeurs, concepteurs, testeurs).
Moyen	Lorsque cela implique le travail de 3 à 4 collaborateurs. (Analystes, développeurs, concepteurs, testeurs).
Bas	Lorsque cela implique le travail de 1 à 2 collaborateurs. (Analystes, développeurs, concepteurs, testeurs).

Risque

Haut	Fonctionnalité qui est difficile à implémenter et dont plusieurs autres fonctionnalités nécessitent l'implémentation de celle-ci. Elle est critique et peut mettre le projet en péril.
Moyen	Fonctionnalité qui demande réflexion (étude) avant d'être implémentée, mais qui ne mettra pas le projet en péril.
Bas	Fonctionnalités qui sont faciles à implémenter. Par exemple, une fonctionnalité similaire qui a déjà été implémentée ou bien qui provient d'une librairie.

Stabilité

Haut	L'équipe de développement est très confiante d'avoir compris la fonctionnalité et est certaine que celle-ci a peu de chance de changer.
Moyen	L'équipe de développement semble avoir compris le contenu de la fonctionnalité, mais il reste un peu de développement à faire avant de l'implémenter de façon finale.
Bas	L'équipe de développement n'est pas certaine d'avoir bien compris le but de la fonctionnalité et suggère une nouvelle analyse.

Priorité

Critique	Fonctionnalité qui devra être implémentée dans les premières itérations. Cette fontionnalité doit absolument être fonctionnelle dans la version finale sinon l'application ne répondra pas aux besoins du client.
Important	Fonctionnalité qui est importante pour l'efficacité du logiciel. L'échec d'une telle fonctionnalité peut influencer le niveau de satisfaction du client.
Utile	Fonctionnalité qui est utile aux yeux du client. Ce type de fonctionnalité est « un plus » pour le système mais n'affectera le degré de satisfaction du client à elle seule et ne peut pas retarder le projet.